In [ ]:
```
#Assignment-3 For Loop in Python
```

In [4]:
```
#1. Write a Python program to print all odd and even numbers from 1 to 20.
print("Even numbers from 1 to 20:")
for num in range(1, 21):
    if num % 2 == 0:
        print(num, end=" ")
print("\nOdd numbers from 1 to 20:")
for num in range(1, 21):
    if num % 2 != 0:
        print(num, end=" ")
```

```
Even numbers from 1 to 20:
2 4 6 8 10 12 14 16 18 20
Odd numbers from 1 to 20:
1 3 5 7 9 11 13 15 17 19
```

In [7]:
```
#. Write a Python program to generate all multiples of 12.
print("Multiples of 12 from 1 to 120:")
for num in range(12, 120, 12):
    print(num, end=" ")
```

```
Multiples of 12 from 1 to 120:
12 24 36 48 60 72 84 96 108
```

In [8]:
```
#3. Write a Python program to generate a table of a number provided by the user.
num = int(input("Enter a number to generate its multiplication table: "))
print(f"\nMultiplication Table of {num}:")
for i in range(1, 10):
    print(f"{num} x {i} = {num * i}")
```

```
Multiplication Table of 10:
10 x 1 = 10
10 x 2 = 20
10 x 3 = 30
10 x 4 = 40
10 x 5 = 50
10 x 6 = 60
10 x 7 = 70
10 x 8 = 80
10 x 9 = 90
```

In [9]:
```
#4. Write a Python program to check if a number provided by the user is prime or no
num = int(input("Enter a number to check if it's prime: "))
if num <= 1:
    print(f"{num} is not a prime number.")
else:
    is_prime = True
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            is_prime = False
            break
    if is_prime:
        print(f"{num} is a prime number.")
```

```
    else:
        print(f"{num} is not a prime number.")
```

```
15 is not a prime number.
```

In [10]:
```
#5. Write a Python program to calculate the sum of numbers between a starting and
#ending point provided by the user.
start = int(input("Enter the starting number: "))
end = int(input("Enter the ending number: "))
total = 0
for num in range(start, end + 1):
    total += num

print(f"\nSum of numbers from {start} to {end} is: {total}")
```

```
Sum of numbers from 10 to 20 is: 165
```

In [2]:
```
#6. Write a Python program to calculate the product of numbers between a starting
#and ending point provided by the user.

start = int(input("Enter the starting number: "))
end = int(input("Enter the ending number: "))
product = 1
for num in range(start, end + 1):
    product *= num

print(f"\nProduct of numbers from {start} to {end} is: {product}")
```

```
Product of numbers from 5 to 10 is: 151200
```

In [3]:
```
#7. Write a Python program to generate the Fibonacci sequence up to a specified
#number of terms.

terms = int(input("Enter the number of terms for the Fibonacci sequence: "))
a, b = 0, 1

print("\nFibonacci Sequence:")
for _ in range(terms):
    print(a, end=" ")
    a, b = b, a + b
```

```
Fibonacci Sequence:
0 1 1 2 3 5 8 13 21 34
```

In [6]:
```
#8. Write a Python program to calculate the factorial of a number provided by the
#user.
num = int(input("Enter a number to calculate its factorial: "))
factorial = 1
if num < 0:
    print(" Factorial is not defined for negative numbers.")
elif num == 0:
    print("Factorial of 0 is: 1")
else:
    for i in range(1, num + 1):
        factorial *= i
    print(f"Factorial of {num} is: {factorial}")
```

```
Factorial of 7 is: 5040
```

In [7]:
```
#9. Write a Python program to find the greatest character from the string "python".

text = "python"
greatest_char = max(text)
print(f"The greatest character in '{text}' is: '{greatest_char}'")
```

```
The greatest character in 'python' is: 'y'
```

In [8]:
```
#10.Write a Python program to display all letters except 'm' and 'i' from the strin
#"Dreamer infotech".
text = "Dreamer infotech"
for char in text:
    if char == 'm' or char == 'i':
        continue
    print(char, end='')
```

```
Dreaer nfotech
```

In [9]:
```
#11.Write a Python program to print alternate characters from a given string.

text = input("Enter a string: ")
alternate_chars = text[::2]
print("Alternate characters are:", alternate_chars)
```

```
Alternate characters are: ue1
```

In [10]:
```
#12.Write a Python program to reverse a string entered by the user.
text = input("Enter a string to reverse: ")
reversed_text = text[::-1]
print("Reversed string is:", reversed_text)
```

```
Reversed string is: dnana
```

In [12]:
```
#13.Write a Python program to count the total number of characters in a string
#entered by the user.
text = input("Enter a string: ")
total_characters = len(text)
print("Total number of characters:", total_characters)
```

```
Total number of characters: 15
```

In [13]:
```
#14.Write a Python program to check whether a string entered by the user is a
#palindrome.

text = input("Enter a string: ")
normalized_text = text.replace(" ", "").lower()
if normalized_text == normalized_text[::-1]:
    print(" The string is a palindrome.")
else:
    print(" The string is not a palindrome.")
```

```
The string is not a palindrome.
```

In [15]:
```
#15.Write a Python program that allows the user to search for a character within a
#given string.

text = input("Enter a string: ")
char_to_search = input("Enter the character to search for: ")
```

```
if char_to_search in text:
    print(f" Character '{char_to_search}' found in the string.")
else:
    print(f" Character '{char_to_search}' not found in the string.")
```

```
Character 'h' found in the string.
```

In [16]:
```
#16.Write a Python program to filter out all vowels and consonants from a string
#entered by the user.

text = input("Enter a string: ")
vowels = "aeiouAEIOU"
vowel_list = []
consonant_list = []
for char in text:
    if char.isalpha():  # Check if it's a letter
        if char in vowels:
            vowel_list.append(char)
        else:
            consonant_list.append(char)
print("Vowels found:", ", ".join(vowel_list))
print("Consonants found:", ", ".join(consonant_list))
```

```
Vowels found: e, i, e, o, a, i
Consonants found: D, l, h, N, w, K, n, d, l
```

In [17]:
```
#17.Write a Python program to filter out duplicate characters from a string entered
#the user.

text = input("Enter a string: ")
unique_chars = ""
for char in text:
    if char not in unique_chars:
        unique_chars += char
print("String after removing duplicates:", unique_chars)
```

```
String after removing duplicates: Delhi NwKonda
```

In [18]:
```
#18.Write a Python program to display all possible pairs of 3.
#Example: 1:3, 2:3, 3:3 , 2:1 , 2:2 ,2:3 , 3:1 ,3:2 ,3:3

for i in range(1, 4):
    for j in range(1, 4):
        print(f"{i}:{j}")
```

```
1:1
1:2
1:3
2:1
2:2
2:3
3:1
3:2
3:3
```

In [19]:
```
#19.Write a Python program to generate the pattern of the letter H.

height = 7
```

```
for row in range(height):
    for col in range(5):
        if col == 0 or col == 4:
            print("*", end="")
        elif row == height // 2:
            print("*", end="")
        else:
            print(" ", end="")
    print()
```

```
*   *
*   *
*   *
*****
*   *
*   *
*   *
```

In [20]:
```
#20.Write a Python program to find duplicate letters between two strings.
#Example: In "virat" and "rohit", the duplicate letter is "r".


str1 = input("Enter first string: ")
str2 = input("Enter second string: ")
set1 = set(str1)
set2 = set(str2)
duplicates = set1.intersection(set2)
if duplicates:
    print("Duplicate letters found:", ", ".join(duplicates))
else:
    print("No duplicate letters found.")
```

```
Duplicate letters found: i, r, t
```

In [21]:
```
#21.Write a Python program to display the squares of numbers from 1 to 10.

for num in range(1, 11):
    square = num ** 2
    print(f"Square of {num} is: {square}")
```

```
Square of 1 is: 1
Square of 2 is: 4
Square of 3 is: 9
Square of 4 is: 16
Square of 5 is: 25
Square of 6 is: 36
Square of 7 is: 49
Square of 8 is: 64
Square of 9 is: 81
Square of 10 is: 100
```

In [22]:
```
#22.Given a string text = "python", calculate the sum of the indices of its
#characters without using the range() or len() functions.

text = "python"
index = 0
```

```
sum_indices = 0
for _ in text:
    sum_indices += index
    index += 1
print("Sum of indices:", sum_indices)
```

```
Sum of indices: 15
```

In [23]:
```
#23.Given: text = "python programming"
#Goal: Count how many vowels are in the string.
#Constraint: Do not use indexing (text[i]) or slicing (text[:]).

text = "python programming"
vowels = "aeiouAEIOU"
vowel_count = 0
for char in text:
    if char in vowels:
        vowel_count += 1
print("Number of vowels:", vowel_count)
```

```
Number of vowels: 4
```

In [24]:
```
#24.Given: text = "programming"
#Goal: Print all characters that repeat in the string.

text = "programming"
repeated_chars = []
for char in text:
    if text.count(char) > 1 and char not in repeated_chars:
        repeated_chars.append(char)
print("Repeated characters:", ", ".join(repeated_chars))
```

```
Repeated characters: r, g, m
```

In [25]:
```
#25.Given : 01275623
#Write a Python program to find and print the greatest character in the string.

text = "01275623"
greatest_char = max(text)
print("Greatest character in the string is:", greatest_char)
```

```
Greatest character in the string is: 7
```

In [26]:
```
#26.Given: text = "knowyourself"
#Goal: Find and print the first character that repeats.

text = "knowyourself"
seen = set()
for char in text:
    if char in seen:
        print("First repeating character is:", char)
        break
    seen.add(char)
else:
    print("No repeating character found.")
```

```
First repeating character is: o
```

In [27]:
```
#27.Give : text=”if you think you can not do, you can not show think wisely”
#Goal: Print the alternate words
#Constraint: Do not use space between words more than once .


text = "if you think you can not do, you can not show think wisely"


words = text.split()
alternate_words = words[::2]
result = ' '.join(alternate_words)

print("Alternate words:", result)
```

```
Alternate words: if think can do, can show wisely
```

In [32]:
```
#28.Given: text = "knowyourself"
#Goal: Find and print the alternate characters.

text = "knowyourself"
alternate_chars = text[::2]
print("Alternate characters:", alternate_chars)
```

```
Alternate characters: koyusl
```

In [34]:
```
#29.Take two numbers from the user: start and end. Print a string labeling each
#number in that range as Odd or Even.
#Output_format : 3:Odd 4:Even 5:Odd 6:Even 7:Odd 8:Even 9:Odd


start = int(input("Enter start number: "))
end = int(input("Enter end number: "))
result = ""
for num in range(start, end + 1):
    label = "Even" if num % 2 == 0 else "Odd"
    result += f"{num}:{label} "
print(result.strip())
```

```
1:Odd 2:Even 3:Odd 4:Even 5:Odd 6:Even 7:Odd 8:Even 9:Odd 10:Even
```

In [35]:
```
#30.Find the sum of string “198765456412”.

text = "198765456412"
total = 0
for char in text:
    total += int(char)
print("Sum of digits:", total)
```

```
Sum of digits: 58
```

In [36]:
```
#31.Count how many digits in the string are greater than 5 from text = "1234567890"

text = "1234567890"
count = 0
for char in text:
    if int(char) > 5:
        count += 1
```

```
print("Digits greater than 5:", count)
```

```
Digits greater than 5: 4
```

In [1]:
```
#32.Task: Replace Character in String
#Write a program that takes a string input from the user, then asks for a character
#to replace and the character to replace it with. The program should output the
#modified string where all occurrences of the specified character are replaced by
#the replacement character.



text = input("Enter a string: ")
old_char = input("Enter the character to replace: ")
new_char = input("Enter the replacement character: ")
modified_text = text.replace(old_char, new_char)

print("Modified string:", modified_text)
```

```
Modified string: Good Eorning
```

In [2]:
```
#33.Replace Spaces with Underscores Replace all spaces in a string with underscores

text = input("Enter a string: ")
modified_text = text.replace(" ", "_")

print("Modified string:", modified_text)
```

```
Modified string: Learn_python_with_dev_sir
```

In [3]:
```
#34.Remove Duplicate characters from the string given by the user then print the fi
#output.

text = input("Enter a string: ")
result = ""
seen = set()
for char in text:
    if char not in seen:
        result += char
        seen.add(char)

print("String after removing duplicates:", result)
```

```
String after removing duplicates: Python
```

In [4]:
```
#35.Take string from user and Replace every vowel in the string with an asterisk *.

text = input("Enter a string: ")
vowels = "aeiouAEIOU"
result = ""
for char in text:
    if char in vowels:
        result += "*"
    else:
        result += char
```

```
print("Modified string:", result)
```

```
Modified string: Pyth*n
```

In [5]:
```
#36.Count only words not spaces.
#Entered a string: Hello coders from Success24
#Number of words: 4

text = input("Entered a string: ")
words = text.split()
word_count = len(words)

print("Number of words:", word_count)
```

```
Number of words: 4
```

In [1]:
```
#37.Task: Count how many uppercase and lowercase letters are in a string.

text = input("Enter a string: ")
uppercase_count = 0
lowercase_count = 0
for char in text:
    if char.isupper():
        uppercase_count += 1
    elif char.islower():
        lowercase_count += 1
print("Uppercase letters:", uppercase_count)
print("Lowercase letters:", lowercase_count)
```

```
Uppercase letters: 0
Lowercase letters: 20
```

In [2]:
```
#38.Task: Print all characters from the string that are at odd indices.

text = input("Enter a string: ")
odd_index_chars = text[1::2]

print("Characters at odd indices:", odd_index_chars)
```

```
Characters at odd indices: yhn
```

In [3]:
```
#39.Task: Create a new string by removing all spaces from the input string.
#Enter a string: how are you all
#String without spaces: howareyouall

text = input("Enter a string: ")
no_spaces = text.replace(" ", "")

print("String without spaces:", no_spaces)
```

```
String without spaces: MynameisAnand
```

In [4]:
```
#40.Count how many digits in a string entered by the user.
#text=”sytax_error2806 hai ”

text = input("Enter a string: ")
digit_count = 0
```

```
for char in text:
    if char.isdigit():
        digit_count += 1

print("Number of digits in the string:", digit_count)
```

```
Number of digits in the string: 4
```

In [5]:
```
#41.Count only alphabets (both uppercase and lowercase).
#Enter a string: this123 i am
#Number of letters: 7

text = input("Enter a string: ")
letter_count = 0
for char in text:
    if char.isalpha():
        letter_count += 1

print("Number of letters:", letter_count)
```

```
Number of letters: 7
```

In [6]:
```
#42.You are managing a simple banking system that tracks the balance at the end of
#each day over 10 days. Each day, the balance increases by 100 units starting from
#on day 1, 200 on day 2, and so on. You want to print the current day's balance alo
#the previous day's balance. For day 1, the previous day's balance is 0.
#Sample output:
#Day 1: Balance = 100, Previous Day Balance = 0
#Day 2: Balance = 200, Previous Day Balance = 100
#Day 3: Balance = 300, Previous Day Balance = 200
#...
#Day 10: Balance = 1000, Previous Day Balance = 900

previous_balance = 0

for day in range(1, 11):
    current_balance = day * 100
    print(f"Day {day}: Balance = {current_balance}, Previous Day Balance = {previou
    previous_balance = current_balance
```

```
Day 1: Balance = 100, Previous Day Balance = 0
Day 2: Balance = 200, Previous Day Balance = 100
Day 3: Balance = 300, Previous Day Balance = 200
Day 4: Balance = 400, Previous Day Balance = 300
Day 5: Balance = 500, Previous Day Balance = 400
Day 6: Balance = 600, Previous Day Balance = 500
Day 7: Balance = 700, Previous Day Balance = 600
Day 8: Balance = 800, Previous Day Balance = 700
Day 9: Balance = 900, Previous Day Balance = 800
Day 10: Balance = 1000, Previous Day Balance = 900
```

In [ ]:
```

```